بسم الله الرحمن الرحيم

Fuzzy Control Course

Lec 7

ANFIS

DR. M. Arafa

22/11/2016

Computational Intelligence Controllers

Controllers that used the following Fields :

1. Fuzzy Logic (FL) / Fuzzy Control (FC)
2. Artificial Neural Networks (ANN)
3. ANFIS
4. Evolutionary Computation

Computational Intelligence Controllers

4. Evolutionary Computation:

Heuristic Algorithms Based on the principles of Darwinian evolution observed in nature.

Example of these algorithms:

- genetic algorithm (GA)
- Differential Evolution Algorithm (DE)
- Particle swarm optimization (PSO)
- Fish Swarm Algorithm (FSA)
- Ant Colony Optimization Algorithm (ACO)
- Artificial Bee Colony Algorithm (ABC)
- Shuffled frog-leaping algorithm (SFLA)

ANFIS based on TSK fuzzy inference system

- The ANFIS is a fuzzy system which is modelled in the form of the artificial neural network (ANN) so that a learning algorithm can be used to train the system.
- ANFIS was introduced by **Jang in 1993**
- ANFIS stands for Adaptive Neuro Fuzzy Inference System or, Adaptive Network-based Fuzzy Inference System.
- TSK fuzzy inference system is simple in computation and easy to be combined with optimizing and self-adapting methods, so that ANFIS based on TSK fuzzy inference system is the most type that is commonly used.

Optimizing ANFIS Parameters

- The main objective of the ANFIS is to determine the optimum values of the equivalent fuzzy inference system parameters (of TSK type) by applying a learning algorithm using input-output data sets (training and test).
- The parameters optimization are done in such a way during training process that the error between the desired and the actual output is minimized.
- The parameters to be optimized in ANFIS are the premise (antecedent or IF part) parameters which describe the shape of the MFs, and the consequent (conclusion or THEN part) parameters which describe the overall output of the system.

Optimizing ANFIS Parameters

- The obtained optimum parameters are then used in testing session to calculate the prediction.

- Various methods have been previously proposed to optimize (train) ANFIS parameters. These methods can be divided into two types:

  1- derivative-based methods 2- derivative-free methods

- **Derivative-based methods** include backpropagation (BP), least squares estimate (LSE), and hybrid learning (HL). HL is a combination of LSE and BP.

- **Derivative-free methods** include other evolutionary computation algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), differential evolution (DE), shuffled frog leaping Algorithm (SFLA), artificial bee colony algorithm (ABC).

ANFIS based on TSK fuzzy inference system

EX1: ANFIS model for two inputs (x_1 , x_2) , two MFs for each input are used. MFS (A1 , A2) for x_1 and MFS (B1 , B2) for x_2. Four possible rules are used, these rules are:

R_1 : IF x_1 is $A1$ AND x_2 is $B1$ THEN $y_1 = p_1 x_1 + q_1 x_2 + r_1$

R_2 : IF x_1 is $A1$ AND x_2 is $B2$ THEN $y_2 = p_2 x_1 + q_2 x_2 + r_2$

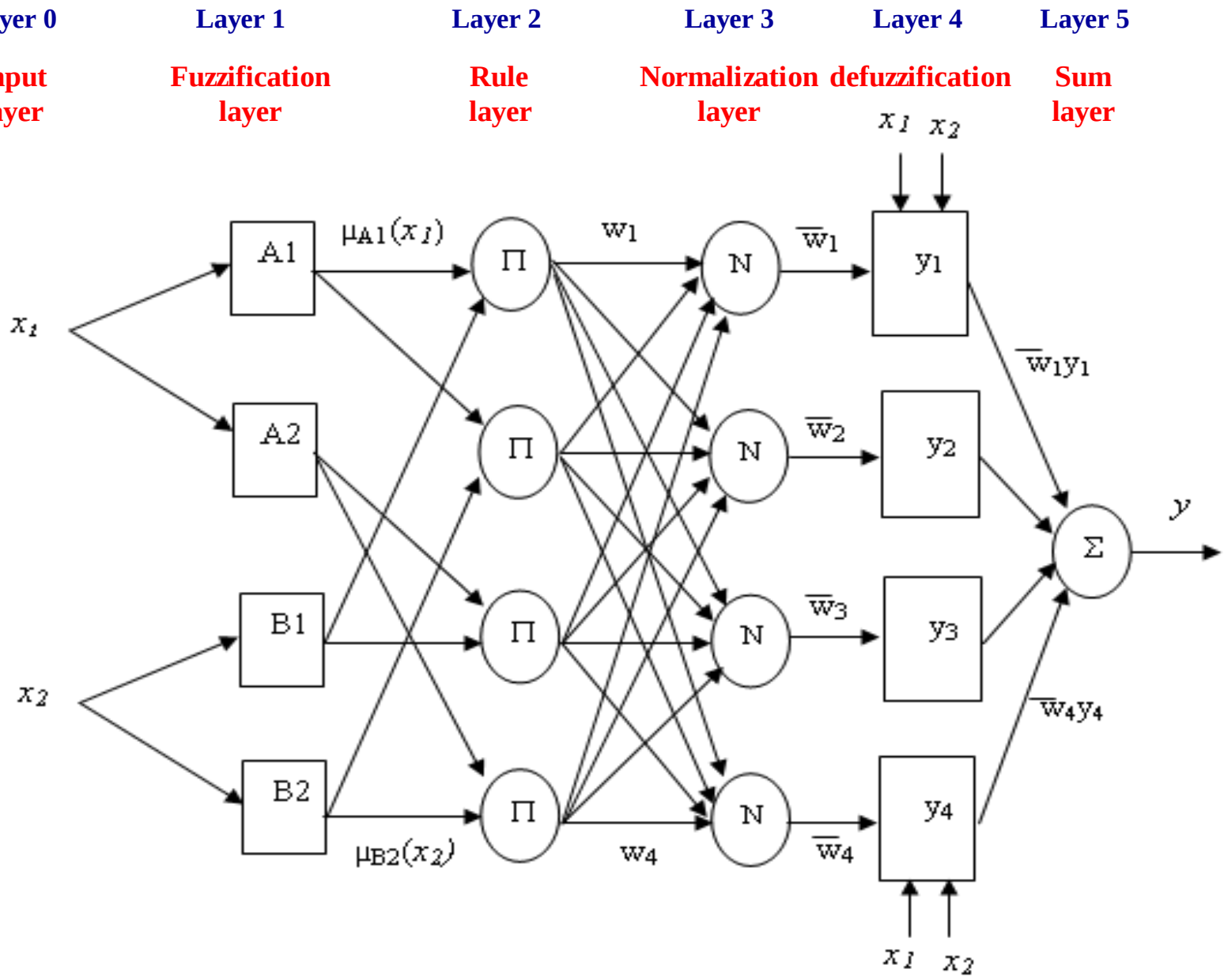
R_3 : IF x_1 is $A2$ AND x_2 is $B1$ THEN $y_3 = p_3 x_1 + q_3 x_2 + r_3$

R_4 : IF x_1 is $A2$ AND x_2 is $B2$ THEN $y_4 = p_4 x_1 + q_4 x_2 + r_4$

- The table of rules for the TSK-fuzzy model:

x_1 \ x_2	**B1**	**B2**
A1	y_1	y_2
A2	y_3	y_4

- The following figure represents the ANFIS structure in the form of the ANN.



The ANFIS structure with two inputs, four rules and one output

ANFIS based on TSK fuzzy inference system

The ANFIS structure shown in the previous figure is a multi-layered neural network, these layers are:

Layer 0

This layer is called **input layer** as the inputs are applying to the system, it does not perform anything else.

Layer 1

This layer is called **fuzzification layer** as the fuzzification process is carried out in this layer by applying the inputs to the membership functions (MFs) and produce a measure of the degree of belonging of each input to a fuzzy MF. This measure is called degree of membership (membership grade) which takes values from the range [0 , 1] and usually denoted to it by μ.

ANFIS based on TSK fuzzy inference system

Layer 2

This layer is called **rule layer** as it executes the fuzzy AND of the antecedent part of the fuzzy rules. The minimum or product operators perform the fuzzy AND. The output of each node represents the fire strength (power) of the rule.

Layer 3

This layer is called **normalization layer**. The output of each node is the ratio of the firing strength of the i^{th} rule to the sum of all firing strengths rules.

$$\bar{w}_i = \frac{w_i}{w_1+w_2+w_3+w_4} \quad , \quad i = 1\,,2\,,3\,,4$$

ANFIS based on TSK fuzzy inference system

Layer 4

This layer is called **defuzzification layer** as it executes the consequent part of the fuzzy rules.
the output of each node in this layer is the product of the normalized firing strength rule and its corresponding linear function in the consequent part.

Layer 5

This layer is called **sum layer** as it computes the total crisp output of the fuzzy system by summing the outputs of layer 4 (this implied the weighted average method of defuzzification).

Defuzzification using Weighted Average Method:

$$y^{crisp} = \frac{w_1 y_1 + w_2 y_2 + w_3 y_3 + w_4 y_4}{w_1 + w_2 + w_3 + w_4} = \bar{w}_1 y_1 + \bar{w}_2 y_2 + \bar{w}_3 y_3 + \bar{w}_4 y_4$$

Notes

- The parameters to be optimized (or tuned) in ANFIS are the premise (IF part) parameters and the consequent (THEN part) parameters .
- The training (learning) process is done to get the best values of these parameters.
- No. of premise parameters = No. of control parameters for MF x total No. of MFs
- No. of consequent parameters = (No. of inputs + 1) x No. of rules
- The total No. of ANFIS parameters =

  No. of premise parameters + No. of consequent parameters

EX:

- Consider the ANFIS model has 3 inputs and each input has 5 triangular MFs.

Determine the followings:

- The total No. of rules
- the total No. of ANFIS parameters

Solution

- The total No. of rules = 5 x 5 x 5 = 125 rule
- The triangular MF has 3 control parameters.
- No. of premise parameters = No. of control parameters for MF x total No. of MFs

  = 3 x 3 x 5 = 45 parameter
- No. of consequent parameters = (No. of inputs + 1) x No. of rules

  = (3 + 1) x 125 = 500 parameter
- The total No. of ANFIS parameters

  = No. of premise parameters + No. of consequent parameters

  = 45 + 500 = 545 parameter

ANFIS based on TSK fuzzy inference system

- **EX2:** ANFIS model for two input (x_1 , x_2) , three MFs (A1 , A2 , A3) are used for input x_1 and two MFS (B1 , B2) are used for x_2. Six possible rules are used, these rules are:

R_1 : IF x_1 is $A1$ AND x_2 is $B1$ THEN $y_1 = p_1 x_1 + q_1 x_2 + r_1$

R_2 : IF x_1 is $A1$ AND x_2 is $B2$ THEN $y_2 = p_2 x_1 + q_2 x_2 + r_2$

R_3 : IF x_1 is $A2$ AND x_2 is $B1$ THEN $y_3 = p_3 x_1 + q_3 x_2 + r_3$

R_4 : IF x_1 is $A2$ AND x_2 is $B2$ THEN $y_4 = p_4 x_1 + q_4 x_2 + r_4$

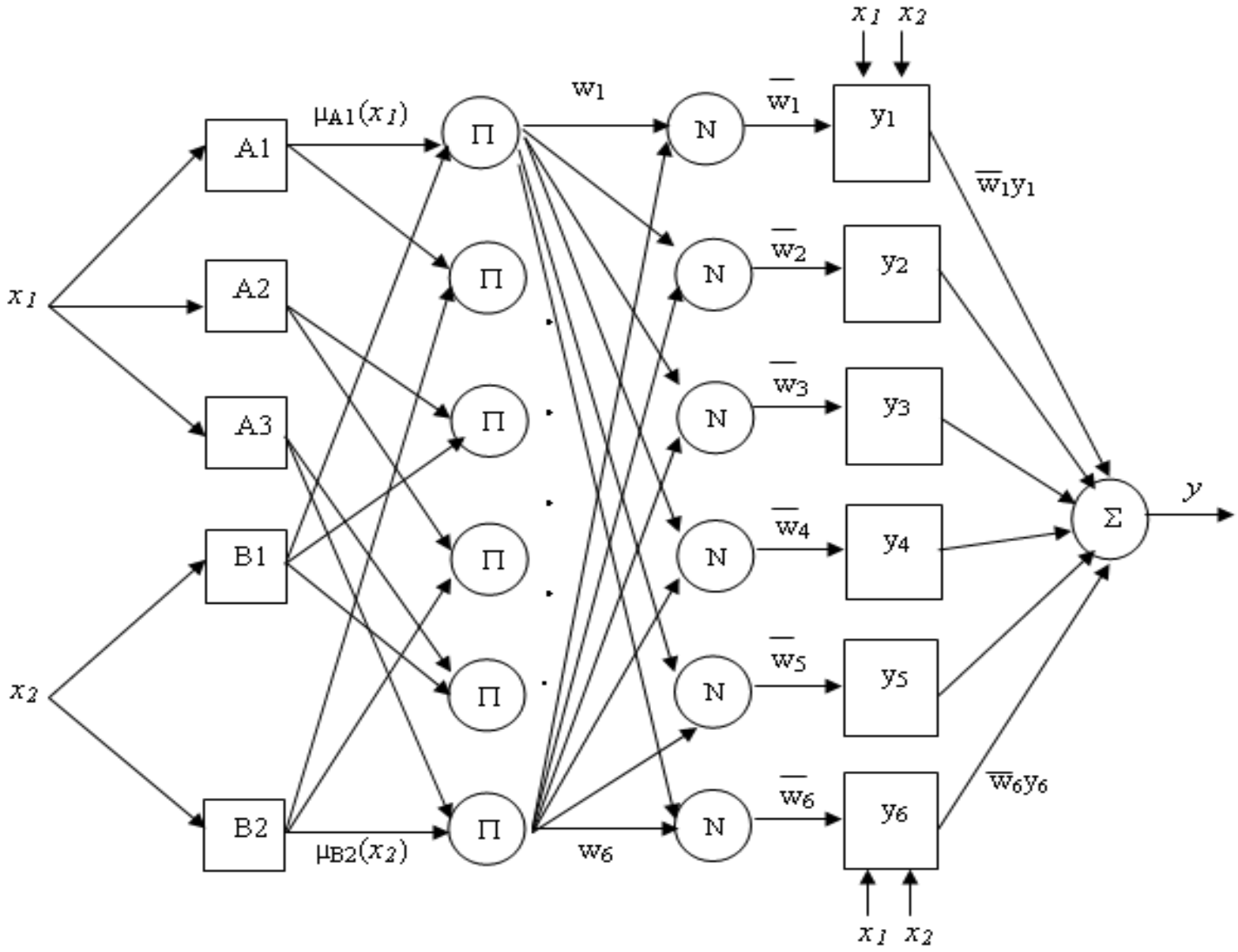
R_5 : IF x_1 is $A3$ AND x_2 is $B1$ THEN $y_5 = p_5 x_1 + q_5 x_2 + r_5$

R_6 : IF x_1 is $A3$ AND x_2 is $B2$ THEN $y_6 = p_6 x_1 + q_6 x_2 + r_6$

- The table of rules for the TSK-fuzzy model:

x_1 \ x_2	*B1*	*B2*
A1	y_1	y_2
A2	y_3	y_4
A3	y_5	y_6

- The following figure represents the ANFIS structure in the form of the ANN.



The ANFIS structure with two inputs, six rules and one output

ANFIS based on TSK fuzzy inference system

- **EX3:** ANFIS model for two input (x_1 , x_2) , three MFs for each input are used. MFS (A1 , A2 , A3) for x_1 and MFS (B1 , B2 , B3) for x_2. Nine possible rules are used, these rules are:

R_1 : IF x_1 is $A1$ AND x_2 is $B1$ THEN $y_1 = p_1 x_1 + q_1 x_2 + r_1$

R_2 : IF x_1 is $A1$ AND x_2 is $B2$ THEN $y_2 = p_2 x_1 + q_2 x_2 + r_2$

R_3 : IF x_1 is $A1$ AND x_2 is $B3$ THEN $y_3 = p_3 x_1 + q_3 x_2 + r_3$

R_4 : IF x_1 is $A2$ AND x_2 is $B1$ THEN $y_4 = p_4 x_1 + q_4 x_2 + r_4$

R_5 : IF x_1 is $A2$ AND x_2 is $B2$ THEN $y_5 = p_5 x_1 + q_5 x_2 + r_5$

R_6 : IF x_1 is $A2$ AND x_2 is $B3$ THEN $y_6 = p_6 x_1 + q_6 x_2 + r_6$

R_7 : IF x_1 is $A3$ AND x_2 is $B1$ THEN $y_7 = p_7 x_1 + q_7 x_2 + r_7$

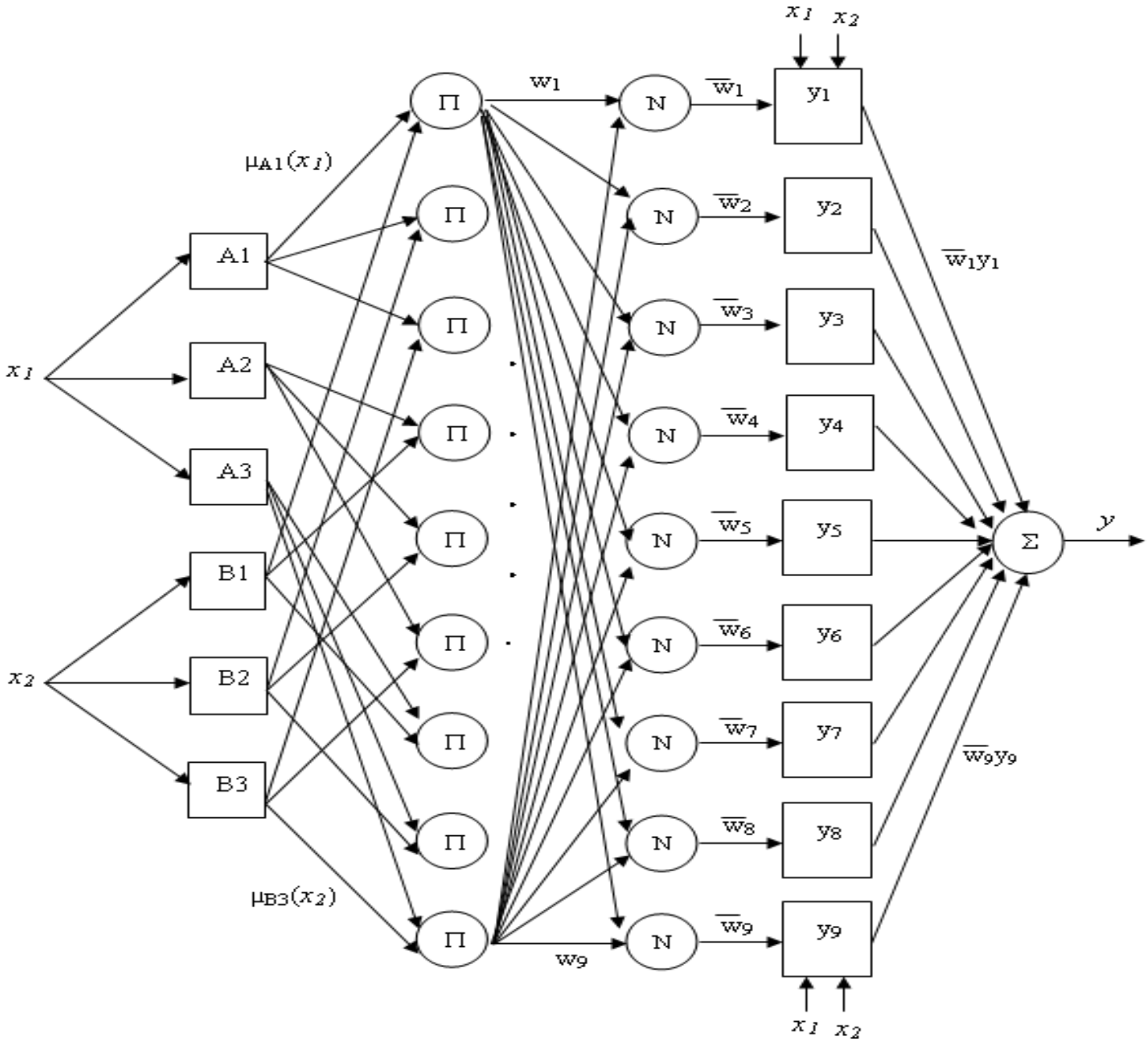
R_8 : IF x_1 is $A3$ AND x_2 is $B2$ THEN $y_8 = p_8 x_1 + q_8 x_2 + r_8$

R_9 : IF x_1 is $A3$ AND x_2 is $B3$ THEN $y_9 = p_9 x_1 + q_9 x_2 + r_9$

- The table of rules for the TSK-fuzzy model:

x_2 / x_1	*B1*	*B2*	*B3*
A1	y_1	y_2	y_3
A2	y_4	y_5	y_6
A3	y_7	y_8	y_9

- The following figure represents the ANFIS structure in the form of the ANN.



The ANFIS structure with two inputs, nine rules and one output

MATLAB Code Example

EX4: Write a MATLAB code to design an ANFIS model to approximate the following function:

$$f(x_1, x_2) = 3x_1 + 2x_2$$

where x_1 and $x_2 \in [-10, 10]$.

Show the best values of the optimizing parameters after leaning or training is done.

Consider the following Specs.:

- Use 100 training data pairs of input and output.
- Use 3 MFs of triangular type for the inputs x_1 and x_2.
- Use the backpropagation algorithm as a learning (or optimization) algorithm.
- The no. of iterations (or epochs) in the training process = 1000.

MATLAB Code Example

Note that:

When we use 3 MFs of triangular type for the inputs x_1 and x_2 :

- The no. of rules = 3 x 3 = 9.
- The no. of ANFIS parameters to be optimized = 27 + 27 = 54.
  - 27 for the antecedent parameters (each triangular MF has 3 tuning parameters).
  - 27 for the consequent parameters (each rule has 3 tuning parameters).

The MATLAB Code

```
                % First, we randomly obtain 100 training data pairs of input and output
                % use the following form of random distribution
                % x_i = x_i^L + ( x_i^U – x_i^L ).*rand (No. of required points , 1);
>> x1 = -10 + (10+10).*rand(100,1); % x1 is a column vector with dimension 100x1
>> x2 = -10 + (10+10).*rand(100,1); % x2 is a column vector with dimension 100x1
>> y = 3.*x1+2.*x2;          % y is a column vector with dimension 100x1
>> x = [x1 x2 y];            % the training data set
>> n = 100;                  % no. of samples of the training data set
>> numEpochs = 1000;
>> numMFs = 3;
>> mfType = ' trimf ';     % we can use 'gaussmf ' or ' trapmf ' or 'gbellmf '
                          % build ANFIS model with the default initialization of parameters
                          % Save initial ANFIS model as fismat1
>> fismat1= genfis1(x,numMFs,mfType);
                          % Or we can use: fismat1=genfis1(x,[3 5],char('trimf ','gaussmf '));
```

The MATLAB Code

```
                    % Train the initial ANFIS model over the training data set x
                    % Save the trained ANFIS model as fismat2
>> [fismat2,trn_mse,tst_mse] = anfis(x,fismat1,numEpochs,NaN,x,0);
 % NaN represent the default values of training options like training epoch number,
 % training error goal,initial step size, step size decrease rate and step size increase rate
 % trn_mse is avector contains the values of mean square errors during the epochs of training process
% tst_mse is avector contains the values of mean square errors during testing process
>> trn_out  = evalfis(x(:,1:2),fismat2);     % the values of outputs for the trained ANFIS
>> [x(:,3)  trn_out]          % view the desired output and the trained ANFIS output
>> AMSE = mean(trn_mse);     % the average of mean square errors over the 1000 epochs.
>> epoch = 1:numEpochs;
>> plot(epoch, trn_mse)       % curve for mean square error vs. epochs
>> yy = [1 : n ];                    % Data Set Index
>> plot(yy,x(:,3),'x',yy,trn_out,'o')
```